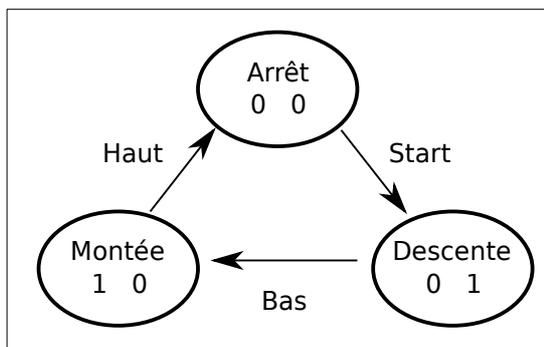
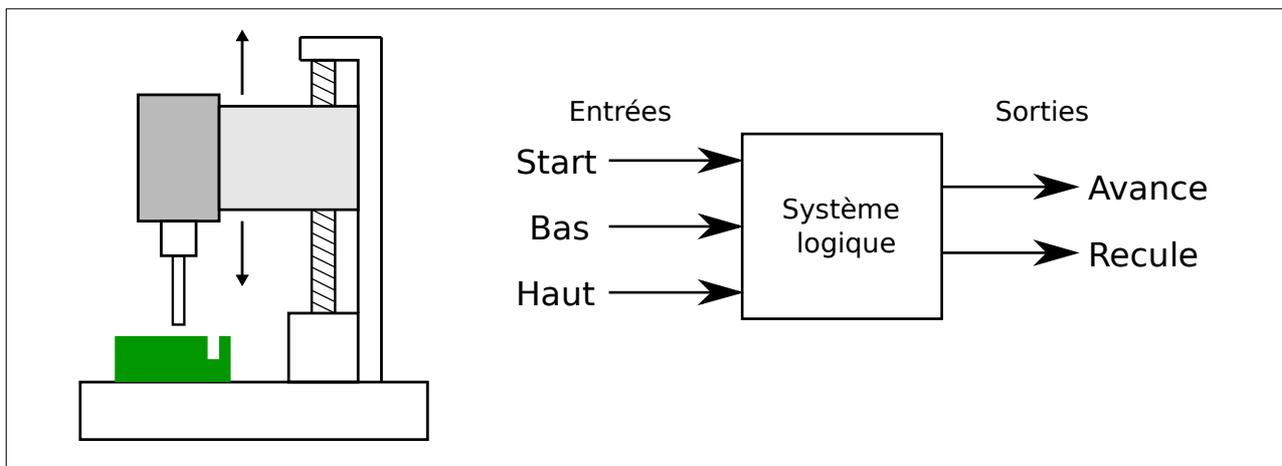


En étudiant les systèmes logiques, on voit d'abord les systèmes combinatoires, caractérisés par le fait que l'état des sorties ne dépend que de l'état présent des entrées. Mais de tels systèmes ne peuvent résoudre qu'une petite partie des problèmes qui se présentent. Même une commande automatique très simple ne peut être réalisée par un système combinatoire. On appelle **systèmes séquentiels** ceux qui ne sont pas combinatoires. L'état des sorties ne dépend pas seulement de l'état présent des entrées, mais aussi de l'état passé du système. Ils contiennent de éléments de mémorisation, qui peuvent être réalisés avec des portes logiques ou des bascules.

Les systèmes séquentiels peuvent être décrits par un **graphe d'état**. Il est composé d'un certain nombre d'**états** (représenté généralement par un cercle ou un ovale) et de **transitions** (représentées par des flèches), allant chacune d'un état vers un autre état.

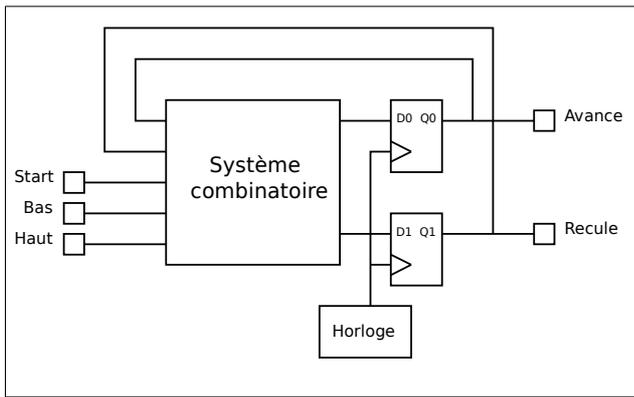
Chaque état est décrit par son **nom** (ou par son numéro). Il correspond aussi à une **valeur** bien déterminée des **sorties du système**. Chaque transition est associée à une **condition logique** sur les **entrées du système**.

Prenons un exemple concret. On cherche à automatiser le mouvement de descente puis de montée d'une perceuse. Un moteur, pouvant être activé dans les deux sens, va faire descendre puis monter le bloc de perçage (comportant le moteur entraînant la mèche). Deux interrupteurs de fin de course sont placés en bas et en haut, pour terminer respectivement la descente et la montée. Le bouton-poussoir *Start* permet à l'ouvrier de démarrer le perçage.



La figure ci-contre donne le graphe d'état. On y trouve trois états: Arrêt, Descente et Montée. Les deux valeurs binaires associées à chaque état correspondent aux sorties **Avance** et **Retour** qui commandent le moteur.

Les transitions permettent le passage d'un état à l'autre. Elles sont associées à une condition sur les entrées.



On peut réaliser une **machine d'état**, appelée aussi **système logique séquentiel synchrone**, en utilisant des **bascules** pour mémoriser l'état, un **système combinatoire** pour calculer l'**état futur** en fonction de l'état présent et des entrées, et un autre système combinatoire pour calculer la **valeur des sorties** en fonction de l'état présent.

Dans notre problème de perceuse, le système combinatoire de sortie est

inexistant, parce qu'on peut coder les états directement avec les valeurs de sorties. On a donc le schéma ci-dessus.

Il reste à déterminer la **table de vérité** du système combinatoire. Comme il a cinq entrées, la table de vérité aura 32 lignes (2^5 à la puissance 5). Plutôt que de chercher à la remplir dans l'ordre binaire habituel, on va examiner successivement chaque état présent, et voir les états futurs possibles, en fonction des entrées. Ainsi, à l'état arrêt correspond deux états futurs possibles: l'état arrêt (on ne change pas d'état) ou l'état descente, lorsqu'on a pressé sur *Start*. Les deux premières lignes représentent ces deux cas. Comme les valeurs de *Haut* et de *Bas* n'interviennent pas dans cet état, on les représente par X. Chacune de ces deux lignes correspond donc à quatre lignes de la table de vérité.

En étudiant les deux autres états, on peut compléter le tableau par les quatre lignes suivantes. Finalement, la valeur 11 qui ne représente aucun état dans notre graphe, doit être associée à l'état futur *Arrêt* pour éviter que la machine d'état ne se bloque dans une état indéterminé (état puis).

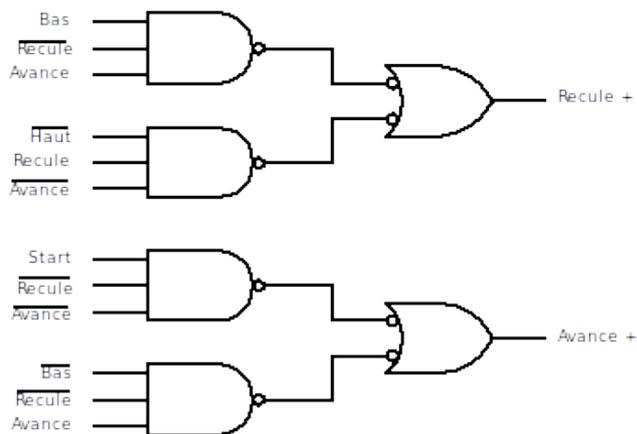
Start	Bas	Haut	Reculé	Avance	Reculé+	Avance+
0	X	X	0	0	0	0
1	X	X	0	0	0	1
X	0	X	0	1	0	1
X	1	X	0	1	1	0
X	X	0	1	0	1	0
X	X	1	1	0	0	0
X	X	X	1	1	0	0

A partir de cette table décrivant les états, on pourrait écrire une table de vérité sous sa forme habituelle et appliquer la méthode des table de *Karnough* pour trouver le schéma logique du système combinatoire. En fait, cette table met justement en évidence les simplifications et on peut en déduire directement les équations logiques des sorties :

$$\text{Reculé+} = (\text{Bas} \cdot \overline{\text{Reculé}} \cdot \text{Avance}) + (\overline{\text{Haut}} \cdot \text{Reculé} \cdot \overline{\text{Avance}})$$

$$\text{Avance+} = (\text{Start} \cdot \overline{\text{Reculé}} \cdot \overline{\text{Avance}}) + (\overline{\text{Bas}} \cdot \overline{\text{Reculé}} \cdot \text{Avance})$$

Le schéma avec des portes Nand est donné ci-dessous :



La réalisation pratique de ce montage nécessiterait au moins cinq circuits intégrés classiques : deux bascules avec un circuit *74HC74*, un oscillateur avec un *NE555* et quelques portes logiques.

Pour simplifier la réalisation, un seul circuit pourrait suffire : un microcontrôleur ! Trois pattes seraient utilisées pour les entrées, deux pour les sorties, deux autres pour les alimentations et deux autres pour la remise à zéro et la programmation. Et si un microcontrôleur peut jouer ce rôle, il peut en jouer bien d'autres encore ! Mais pour cela, il faut le programmer...

Voici le schéma correspondant, réalisé avec un microcontrôleur à 8 pattes seulement :

